

REMARKS/ARGUMENTS

This is a complete and timely response to the Office action of January 18, 2007.
Claim 1 has been slightly amended for grammatical reasons, claim 2 has been amended to
end with a period, and the abstract has also been amended to comprise no more than 150
5 words as required. Reconsideration of claims 1-9 is respectfully requested.

1. Background

Claims 1-9 are rejected under 35 U.S.C. 102(e) as being anticipated by Augsburg et
al. U.S. 6,826,747.

10

2. Claim rejections

Concerning claim 1, on page 3 of the current Office action, “initial state, col.5, lines
13-21” is cited by the Examiner as anticipating “setting a trace count value as a initial
value; setting a trigger count value according to the tracing start point;”

15

From the cited text, the applicant understands an “initial state” to be a
synchronizing event (SE) and that a trigger event (TE) is to correspond to the present
“trigger count value”. From additional citations (for example col.9, lines 41-52), it
appears that Augsburg’s “SE counter 120” is suggested to correspond to the present
20 “trace count value”, although this is not specifically stated.

At the end of the same page of the Office action, the Examiner cites “col.8, lines
47-61” to suggest anticipation of “resetting the trigger count value according to the
trace count value while the buffer is full, resetting the tracing value with the initial
25 value, using the microprocessor to start to run the program; and when the
microprocessor finishes running the program, outputting instruction information
recorded in the buffer via the output interface.”

The cited text describes Augsburg’s Fig.2, and in particular the recycling of step
30 215 until the FIFO is no longer full. The text also discloses “if an SE event has

occurred, then in step 202, parameters SE-IAR-PENDING, SE-LR-PENDING, and SE-CTR-PENDING are made equal to 1.” and that these parameters are subsequently and individually each reset to zero.

5 However, the applicant is unable to locate teachings or suggestion of resetting the trigger count value according to the trace count value while the buffer is full. Paragraph [0011] of the present application says: “After each time the microprocessor executes a specific instruction, the method will increase the trace count value by one and then compare the increased trace count value to the trigger count value”, which more or
10 less corresponds to Augsburg’s SE counter 120. **There are no teachings in Augsburg of resetting the TE’s according to the trace count value while the buffer is full.** In fact, the only portion of Augsburg that touches on setting TEs is found in Col.9, lines 1-6 which says that a user sets up TEs to occur according to events designated by a user through a debug circuit 104. Although Augsburg does allow for multiple TEs to be set up,
15 there is no teaching or suggestion of resetting the TE according to the trace count value while the buffer is full. On the other hand, the present invention resets the TE according to the trace count value while the buffer is full to trigger a new trace at exactly the point where the old trace was stopped when the program is rerun following an emptying of the buffer (Paragraph [0033] and Fig.2).

20 Additionally, the applicant is unable to find teachings or suggestion in Augsburg of “**resetting the tracing value with the initial value**” as described in Paragraph [0033] and Fig.2 of the present application. If the Examiner considers Augsburg’s SE counter 120 to be the equivalent to the present tracing value, there is no evidence that the SE counter 120 is reset to any value except incremented as normal. On the other hand, if
25 the Examiner is likening resetting Augsburg’s parameters (step 202, Fig.2) to their initial value to “resetting the tracing value with the initial value”, the resetting of these parameters has absolutely nothing to do with whether or not the buffer is full, instead the parameters are utilized as flags to send appropriate data to the pins after an SE has occurred (Col.8, lines 20-38).

30

The present invention initializes a trace counter and sets a trigger count value. When the incrementing trace counter reaches the trigger count value, subsequent instructions are copied into the buffer. When the buffer becomes full, program execution is halted, contents of the buffer are output, the trigger count value is set to mark the location in the program where execution was halted (trace counter), the trace counter is reinitialized, and execution of the program is again started according to the reinitialized trace counter (Paragraph [0011]), meaning started over. This cyclic procedure enables a complete trace of the program without interfering with program execution, something which known prior art cannot do.

For at least the above reasons, applicant respectfully requests reconsideration of claims 1-9 and that a timely Notice of Allowance be issued in this case.

Sincerely yours,



Date: 05/18/2007

Winston Hsu, Patent Agent No. 41,526
P.O. BOX 506, Merrifield, VA 22116, U.S.A.
Voice Mail: 302-729-1562
Facsimile: 806-498-6673
e-mail : winstonhsu@naipo.com

Note: Please leave a message in my voice mail if you need to talk to me. (The time in D.C. is 12 hours behind the Taiwan time, i.e. 9 AM in D.C. = 9 PM in Taiwan.)